

FASTER DISCRETE COSINE TRANSFORMS USING SCALED TERMS

CROSS-REFERENCE TO RELATED APPLICATION

This application is related to the following co-pending and commonly-
5 assigned patent applications, which are hereby incorporated herein by reference in
their respective entirety:

"FASTER TRANSFORMS USING SCALED TERMS" to Trelewicz et.al.,
having attorney docket no. BLD9-2000-0059US1.

"FASTER TRANSFORMS USING EARLY ABORTS AND PRECISION
10 REFINEMENTS" to Mitchell et al., having attorney docket no. BLD9-2000-0064US1.

BACKGROUND OF THE INVENTION

1. Field of the Invention.

This invention relates in general to data processing, and more particularly to
faster discrete cosine transforms using scaled terms.

2. Description of Related Art.

20 Transforms, which take data from one domain (e.g., sampled data) to another
(e.g., frequency space), are used in many signal and/or image processing
applications. Such transforms are used for a variety of applications, including, but
not limited to data analysis, feature identification and/or extraction, signal correlation,

data compression, or data embedding. Many of these transforms require efficient implementation for real-time and/or fast execution.

Data compression is desirable in many data handling processes, where too much data is present for practical applications using the data. Commonly,

5 compression is used in communication links, to reduce transmission time or required bandwidth. Similarly, compression is preferred in data storage systems, including digital printers and copiers, where "pages" of a document to be printed may be stored temporarily in memory. Here the amount of media space can be substantially reduced with compression. Generally speaking, scanned images, i.e., electronic
10 representations of hard copy documents, are often large, and thus make desirable candidates for compression.

It is well-known in the art to use a discrete cosine transform for data compression. In particular examples referred to herein, the terms images and image processing will be used. However, those skilled in the art will recognize that the
15 present invention is not meant to be limited to processing images but is applicable to processing different data, such as audio data, scientific data, image data, etc.

In combination with other techniques, such as color subsampling, quantization, Huffman coding and run-length coding, the discrete cosine transform can compress a digital color image by a factor of approximately thirty to one with
20 virtually no noticeable image degradation. Because of its usefulness in data compression, the discrete cosine transform is an integral part of several data compression standards used by international standards committees such as the International Standards Organization.

DCT (Discrete Cosine Transform), disseminated by the JPEG committee, is a lossy compression system which reduces data redundancies based on pixel to pixel correlations. Generally, an image does not change very much on a pixel to pixel basis and therefore has what is known as "natural spatial correlation". In natural scenes, correlation is generalized, but not exact. Noise makes each pixel somewhat different from its neighbors. Moreover, signal and data processing frequently needs to convert the input data into transform coefficients for the purposes of analysis. Often only a quantized version of the coefficients are needed (e.g., JPEG/MPEG data compression or audio/voice compression). Many such applications need to be done fast in real time such as the generation of JPEG data for high speed printers.

Generally, an example of a JPEG DCT compression and decompression system may be had by referencing the Encyclopedia of Graphics File Formats, by J. D. Murray and W. vanRyper, pp. 159-171 (1994, O'Reilly & Associates, Inc.). Further description of the draft JPEG standard may be found, for example, in "JPEG Still Image Data Compression Standard," by W. Pennebaker and J. Mitchell, 1993 (Van Nostrand Reinhold, New York) or "Discrete Cosine Transform: Algorithms, Advantages and Applications," by K. Rao and P. Yip, 1990 (Academic Press, San Diego).

The two-dimensional discrete cosine transform is a pair of mathematical equations that transforms one $N_1 \times N_2$ array of numbers to or from another $N_1 \times N_2$ array of numbers. The first array typically represents a square $N \times N$ array of spatially determined pixel values which form the digital image. The second array is an array of discrete cosine transform coefficients which represent the image in the frequency

5

10

15

20

transform for images use eight-by-eight arrays wherein N therefore has a value of eight. Assuming then that N has the value of eight when performing the transforms, where $f(i, j)$ are the values of the pixel array and $F(u, v)$ are the values of the discrete cosine transform coefficients, the equations of the discrete cosine transforms are as follows.

The formula for the 2D discrete cosine transform is given by:

$$F(u, v) = \frac{C_u C_v}{4} \sum_{x=0}^7 \sum_{y=0}^7 f(x, y) \cos\left(\frac{(2x+1)u\pi}{16}\right) \cos\left(\frac{(2y+1)v\pi}{16}\right)$$

where x, y = spatial coordinates in the spatial domain (0, 1, 2, ..., 7); u, v = coordinates in the transform domain (0, 1, 2, ..., 7); $C_u = \frac{1}{\sqrt{2}}$ for $u = 0$, otherwise 1; and $C_v = \frac{1}{\sqrt{2}}$ for $v = 0$, otherwise 1. The separable nature of the 2D DCT is exploited by performing a 1D DCT on the eight columns, and then a 1D DCT on the eight rows of the result. Several fast algorithms are available to calculate the 8-point 1D DCT.

As described above, a DCT compressor comprises mainly two parts. The first part transforms highly correlated image data into weakly correlated coefficients using a DCT transform and the second part performs quantization on coefficients to reduce the bit rate for transmission or storage. However, the computational burden in performing a DCT is demanding. For example, to process a one-dimensional DCT of length 8 pixels requires 13 multiplications and 29 additions in currently

known fast algorithms. As stated above, the image is divided into square blocks of size 8 by 8 pixels, 16 by 16 pixels or 32 by 32 pixels. Each block is often processed by the one-dimensional DCT in row-by-row fashion followed by column-by-column. On the other hand, different block sizes are selected for compression due to different types of input and different quality requirements on the decompressed data.

In the article, "A Fast DCT-SQ Scheme for Images," Trans. IEICE, Vol. E-71, No. 11, pp. 1095-1097, Nov. 1988, Y. Arai, T. Agui, and M. Nakajima proposed that many of the DCT multiplications can be formulated as scaling multipliers to the DCT coefficients. The DCT after the multipliers are factored out is called the scaled DCT.

The scaled DCT is still orthogonal but no longer normalized, whereas the scaling factors may be restored in a subsequent quantization process. Arai, et al. have demonstrated in their article that only 5 multiplications and 29 additions are required in processing an 8-point scaled DCT.

However, there is a need to further increase the speed of the encoder because more than half of the time in the JPEG encoder is spent in the Forward Discrete Cosine Transform (FDCT) code calculating the two-dimensional (2-D) 8x8 block of 8-bit or 12-bit samples. Currently, the 2-D FDCT is calculated by first calculating eight 1-D horizontal DCTs and then calculating another eight 1-D vertical DCTs using the currently fastest known 1-D DCT, which was first described by Arai, Agui, and Nakajima, as mentioned above. The current process takes 29 additions and 5 multiplication to calculate a scaled version of the 1-D FDCT. The scaling constants are the same for each vertical column and can finally be included in the quantization step. This prior solution saved eight multiplications per 1-D FDCT.

However, as stated above, it can be seen that there is a need to provide a faster DCT transform.

It can also be seen that there is a need to provide a method and apparatus for performing discrete cosine transforms with less addition and multiplication steps to

5 increase throughput of an encoder.

SUMMARY OF THE INVENTION

To overcome the limitations in the prior art described above, and to overcome other limitations that will become apparent upon reading and understanding the present specification, the present invention discloses a faster discrete cosine
5 transform that uses scaled terms.

The present invention solves the above-described problems by decreasing the number of multiplications by using scaling terms on the transform constants. Further, after scaling, the terms of the constants are replaced by a series of fewer linear shifts and additions since the constants may be approximated by sums of
10 powers-of-2 after only a few summations. Those skilled in the art will recognize that throughout this specification, the term 'matrix' is used in both its traditional mathematical sense and also to cover all hardware and software systems which when analyzed could be equivalently represented as a mathematical matrix.

A method in accordance with the principles of the present invention includes
15 arranging discrete cosine transform equations into at least one collection having at least two discrete cosine transform constants, scaling the discrete cosine transform equations in the at least one collection by dividing each of the discrete cosine transform constants in the collection by one of the discrete cosine transform constants from the at least one collection and representing each of the scaled discrete cosine
20 transform constants with estimated scaled discrete cosine transform constants approximated by sums of powers-of-2.

Other embodiments of a method in accordance with the principles of the invention may include alternative or optional additional aspects. One such aspect of

the present invention is that the method further includes separating an image into at least one block and transforming the block into transformed data by performing matrix multiplication on the discrete cosine transform equations based upon binary arithmetic using the estimated scaled discrete cosine transform constants and performing linear shifts and additions.

Another aspect of the present invention is that the scaling the discrete cosine transform equations in the at least one collection by dividing each of the discrete cosine transform constants in the at least one collection by one of the discrete cosine transform constants from the at least one collection saves multiplications.

Another aspect of the present invention is that the discrete cosine transform constant chosen for scaling the discrete cosine transform equations in the at least one collection is selected according to a predetermined cost function.

Another aspect of the present invention is that the cost function minimizes a number of add operations.

Another aspect of the present invention is that the cost function minimizes a worst case number of add operations.

Another aspect of the present invention is that the cost function minimizes an error per constant resulting from the approximations.

Another aspect of the present invention is that the transforming the block into transformed data further comprises generating at least one set of one dimensional discrete cosine transform equations.

Another aspect of the present invention is that the discrete cosine transform constants are obtained by splitting the discrete cosine transform constants into even and odd terms by obtaining sums and differences of input samples.

Another aspect of the present invention is that the block is an $N_1 \times N_2$ block.

5 Another aspect of the present invention is that $N_1 = N_2 = 8$.

In another embodiment of the present invention, a data compression system is provided. The data compression system includes a discrete cosine transformer for applying a discrete cosine transform to decorrelate data into discrete cosine transform equations, the discrete cosine transform equations having been formed by
10 arranging the discrete cosine transform equations into at least one collection having at least two discrete cosine transform constants, scaling the discrete cosine transform equations in the at least one collection by dividing each of the discrete cosine transform constant in the collection by one of the discrete cosine transform constants from the at least one collection and representing each of the scaled
15 discrete cosine transform constants with estimated scaled discrete cosine transform constants approximated by sums of powers-of-2 and a quantizer for quantizing the transformed data into quantized data to reduce the number of bits needed to represent the transform coefficients.

In another embodiment of the present invention, a printer is provided. The
20 printer includes a memory for storing data, a processor for processing the data to provide a compressed print stream output and a printhead driving circuit for controlling a printhead to generate a printout of the data, wherein the processor applies a discrete cosine transform to decorrelate data into transform coefficients

using discrete cosine equations, the discrete cosine transform equations having been formed by arranging the discrete cosine transform equations into at least one collection having at least two discrete cosine transform constants, scaling the discrete cosine transform equations in the at least one collection by dividing each of the discrete cosine transform constant in the collection by one of the discrete cosine transform constants from the at least one collection and representing each of the scaled discrete cosine transform constants with estimated scaled discrete cosine transform constants approximated by sums of powers-of-2.

In another embodiment of the present invention, an article of manufacture is provided. The article of manufacture includes a program storage medium readable by a computer, the medium tangibly embodying one or more programs of instructions executable by the computer to perform a method for performing faster discrete cosine transform, the method includes arranging discrete cosine transform equations into at least one collection having at least two discrete cosine transform constants, scaling the discrete cosine transform equations in the at least one collection by dividing each of the discrete cosine transform constant in the collection by one of the discrete cosine transform constants from the at least one collection and representing each of the scaled discrete cosine transform constants with estimated scaled discrete cosine transform constants approximated by sums of powers-of-2.

In another embodiment of the present invention, a data analysis system is provided. The data analysis system includes a memory for storing the discrete cosine transform equations having been formed by arranging the discrete cosine

transform equations into at least one collection having at least two discrete cosine transform constants, scaling the discrete cosine transform equations in the at least one collection by dividing each of the discrete cosine transform constant in the collection by one of the discrete cosine transform constants from the at least one collection and representing each of the scaled discrete cosine transform constants with estimated scaled discrete cosine transform constants approximated by sums of powers-of-2 and a transformer for applying the transform equations to perform a discrete cosine transform to decorrelate data into discrete cosine transform coefficients.

10 These and various other advantages and features of novelty which characterize the invention are pointed out with particularity in the claims annexed hereto and form a part hereof. However, for a better understanding of the invention, its advantages, and the objects obtained by its use, reference should be made to the drawings which form a further part hereof, and to accompanying descriptive matter, in which there are
15 illustrated and described specific examples of an apparatus in accordance with the invention.

BRIEF DESCRIPTION OF THE DRAWINGS

Referring now to the drawings in which like reference numbers represent corresponding parts throughout:

Fig. 1 illustrates a block diagram of a JPEG encoder;

5 Fig. 2 illustrates a table that shows the terms C2 and C6;

Fig. 3 is a table that shows the actual values if either C2 or C6 is used to scale the coefficient matrix and the estimates are a ratio of integers;

Fig. 4 shows all possible combinations of dividing the coefficient matrix by C1, C3, C5, and C7;

10 Fig. 5 shows the actual values when C5 is used to scale the coefficient matrix;

Fig. 6 shows the actual values when C7 is used to scale the coefficient matrix;

15 Fig. 7 shows the number of binary adds to generate the unscaled cosine terms within one percent error;

Fig. 8 illustrates a flow chart of the method to design the fast DCT using scaled terms according to the present invention;

Fig. 9 illustrates a printer according to the present invention;

Fig. 10 illustrates a data analyzing system according to the present invention;

20 and

Fig. 11 illustrates another data analyzing system according to the present invention.

DETAILED DESCRIPTION OF THE INVENTION

In the following description of the exemplary embodiment, reference is made to the accompanying drawings which form a part hereof, and in which is shown by way of illustration the specific embodiment in which the invention may be practiced.

5 It is to be understood that other embodiments may be utilized as structural changes may be made without departing from the scope of the present invention.

The present invention provides a faster discrete cosine transform that uses scaled terms. The present invention decreases the number of multiplications by using scaled terms for the transform constants. Further, after scaling, the ratios of
10 the constants are replaced by a series of linear shifts and additions since the constants may be approximated by sums of powers-of-2 after only a few summations.

Fig. 1 illustrates a block diagram of a JPEG encoder 100. In Fig. 1, digital image data 110 is divided up into 8 by 8 pixel blocks 112. Then, the discrete cosine transform (DCT) of each block is calculated 120. The discrete cosine transform
15 (DCT) helps separate the data into parts (or spectral sub-bands) of differing importance (e.g., with respect to an image's visual quality). The DCT is similar to the discrete Fourier transform: it transforms a signal from the spatial domain to the frequency domain.

20 A quantizer 130 rounds off the DCT coefficients according to the quantization matrix. This step produces the "lossy" nature of JPEG, but allows for large compression ratios. There is a tradeoff between image quality and degree of quantization. A large quantization step size can produce unacceptably large image

distortion. This effect is similar to quantizing Fourier series coefficients too coarsely; large distortion would result. Unfortunately, finer quantization leads to lower compression ratios. The question is how to quantize the DCT coefficients most efficiently. Because of human eyesight's natural high frequency roll-off, these frequencies play a less important role than low frequencies. This lets JPEG use a much higher step size for the high frequency coefficients, with little noticeable image deterioration. The quantizer coefficient output may then be encoded by an optional entropy encoder 140 to produce a compressed data stream 150 to an output file. Note that for JPEG the entropy encoder is not optional, but other similar data compression systems could be designed without the CPU cycles required by the entropy encoder.

However, as described below, there is a need to provide a method and apparatus for performing discrete cosine transforms with less multiplication steps to increase throughput of the encoder 100. As will be described, the method according to the present invention saves multiplications in the brute force equations by scaling the coefficient matrix. Each separable subgroup is scaled independently of the other collections. Within each collection, the remaining multiplications are replaced by simple shifts and adds. The scaling terms are chosen according to various cost functions. The preferred embodiment uses cost functions that minimize the number of adds and that minimize the worst case number of adds. However, those skilled in the art will recognize that alternate cost functions could choose how much error is allowed per coefficient. Moreover, those skilled in the art will recognize that the

inverse DCT (IDCT) can be implemented using the same method so that the same number of operations are used.

Let the $C_n = \cosine(n * \pi/16)$. Let $f(x)$ for $x = 0$ to 7 be the input samples. Let $F(u)$ for $u=0$ to 7 be proportional to the 1-D transform coefficients. Let $S(u)$ for $u=0$ to 7 be equal to the 1-D transform coefficients. Let s_{xy} be $f(x) + f(y)$ and d_{xy} be $f(x) - f(y)$ for $x, y = 0$ to 7 . Note the calculations for $F(u)$ below ignore some constants that end up in the quantization terms. The first step in the 1-D DCT is to split the coefficients into the even and odd terms by taking sums and differences of the eight input samples.

$$\begin{aligned} s_{07} &= f(0) + f(7) \\ s_{16} &= f(1) + f(6) \\ s_{25} &= f(2) + f(5) \\ s_{34} &= f(3) + f(4) \\ d_{07} &= f(0) - f(7) \\ d_{16} &= f(1) - f(6) \\ d_{25} &= f(2) - f(5) \\ d_{34} &= f(3) - f(4) \end{aligned}$$

Then, the sums and differences of the sums are calculated:

$$\begin{aligned} s_{0734} &= s_{07} + s_{34} \\ s_{1625} &= s_{16} + s_{25} \\ d_{0734} &= s_{07} - s_{34} \\ d_{1625} &= s_{16} - s_{25} \end{aligned}$$

Finally we take sums and differences of the sums again to calculate $F(0)$ and $F(4)$.

$$\begin{aligned} F(0) &= s_{0734} + s_{1625} \\ F(4) &= s_{0734} - s_{1625} \\ 2 * S(0) &= C_4(s_{0734} + s_{1625}) \\ 2 * S(4) &= C_4(s_{0734} - s_{1625}) \end{aligned}$$

Those skilled in the art will recognize that the equations above have been derived similar to the Arai, Agui, and Nakajima algorithm. The F(2) and F(6) can be calculated from these terms as:

$$\begin{aligned}
 5 \quad & F(2) = C2 * d0734 + C6 * d1625 \\
 & F(6) = C6 * d0734 - C2 * d1625 \\
 & 2 * S(2) = C2 * d0734 + C6 * d1625 \\
 & 2 * S(6) = C6 * d0734 - C2 * d1625
 \end{aligned}$$

10 Fig. 2 illustrates a table 200 that shows the C2 and C6 terms divided by C2 or C6. C2 is equal to 0.923880, C2/C2 is equal to 1.000000, C2/C6 is equal to $1 + \sqrt{2}$, C6 is equal to 0.382683, C6/C2 is equal to $\frac{1}{1 + \sqrt{2}}$, and C6/C6 is equal to 1.000000.

Next, the utility of scaling the constants in the matrix by C6 or C2 is investigated.

Fig. 3 is a table 300 that shows the actual values 310 if either C2 or C6 is used to scale the constants in the matrix and the estimates 312 are a ratio of integers. Fig. 3 shows the error 314 due to each of the approximations. C2 was chosen because the greatest error occurs in the harmonic with the least power (or the smallest contribution to the overall output).

20 Since F(2) is much more likely to occur, it is now exactly calculated. So the equations become:

$$\begin{aligned}
 & F(2) = 12 * d0734 + 5 * d1625 = d1625 + (d0734 + d1625 + (d0734 << 1)) << 2 \\
 & F(6) = 5 * d0734 - 12 * d1625 = d0734 + (d0734 - d1625 - (d1625 << 1)) << 2 \\
 & (24/C2) * S(2) = d1625 + (d0734 + d1625 + (d0734 << 1)) << 2 \\
 25 \quad & (24/C2) * S(6) = d0734 + (d0734 - d1625 - (d1625 << 1)) << 2
 \end{aligned}$$

Equation 1 shows the remaining equations for the odd constants expressed in

matrix notation.

$$\begin{bmatrix} F(1) \\ F(3) \\ F(5) \\ F(7) \end{bmatrix} = \begin{bmatrix} +C1 & +C3 & +C5 & +C7 \\ +C3 & -C7 & -C1 & -C5 \\ +C5 & -C1 & +C7 & +C3 \\ +C7 & -C5 & +C3 & -C1 \end{bmatrix} \begin{bmatrix} d07 \\ d16 \\ d25 \\ d34 \end{bmatrix} \quad \text{Equation 1}$$

5 In Equation 1, F(1), F(3), F(5) and F(7) are a vector related to unquantized 1-D DCT transform coefficients. Note: for $u = (1, 3, 5, 7)$, $2 * S(u) = F(u)$. If the solutions for these four coefficients are found using a "brute force" approach, then 12 additional additions and 16 multiplications would be necessary. However, if the matrix of coefficients is divided by one of the constants in the matrix, then the
10 number of multiplications necessary is reduced by four. This linear system is shown in Equation 2 below.

$$\frac{1}{C5} * \begin{bmatrix} F(1) \\ F(3) \\ F(5) \\ F(7) \end{bmatrix} = \begin{bmatrix} +C1/C5 & +C3/C5 & +1 & +C7/C5 \\ +C3/C5 & -C7/C5 & -C1/C5 & -1 \\ +1 & -C1/C5 & +C7/C5 & +C3/C5 \\ +C7/C5 & -1 & +C3/C5 & -C1/C5 \end{bmatrix} \begin{bmatrix} d07 \\ d16 \\ d25 \\ d34 \end{bmatrix} \quad \text{Equation 2}$$

15 Equation 2 shows that the linear set now only involves 12 additions, 12 multiplications and one division. However, the constant C5 in the division can be included in the quantization numbers once per image. Meanwhile, the benefits of only 12 multiplications instead of 16 are reaped every 8 by 8 block of samples. Furthermore, when the constants of the matrix were scaled by one of the terms, all
20 of the terms became approximations of sums of powers-of-2 within a few successive approximations. This is very beneficial because a multiplication which normally is

considered to necessitate more than 5 CPU cycles now becomes a shift and add with a maximum of 3 shifts and 2 additions. Fig. 4 shows all possible combinations of dividing the constants in the matrix by C1, C3, C5, and C7.

Below, the utility of scaling the constants in the matrix by C5 or by C7 is investigated. Fig. 5 is a table 500 showing the actual values 510 if C5 is used to scale the constants of the matrix. The estimates 512 are in sums of powers-of-2, and the error 514 due to the approximation is also stated. C5 was chosen because of the continuity in the terms. It was also chosen because the greatest error occurs in the harmonic with the least power (or the smallest contribution to the overall output). The constant C7 is also a good candidate because the calculations fit inside of 16 bits. This optimization is shown in table 600 of Fig. 6.

Those skilled in the art will recognize the above scaling is not meant to be an optimal solution for minimizing error. Nevertheless, an exhaustive search may be performed to find the optimal number to scale the matrix. Also, the scaling factor need not be a function of $\cos(n * \pi/16)$. As shown for the C2 and C6 cases, the best answer was a ratio of integers. For comparison purposes, Fig. 7 is a table 700 showing the number of binary adds and used to calculate the unscaled cosine terms within one percent error.

In summary, the scaled terms approach fundamentally decreases the number of multiplications because the scaling on the quantization values is done once per image. However, the benefits are reaped every 8x8 block, or more generally, $N_1 \times N_2$ block. Second, after scaling, the ratios of coefficients are very nearly sums of powers-of-2 after only a few summations. Thus, multiplications can be replaced by a

series of linear shifts and additions. In some cases, these terms may be further converted into integer terms. This will provide superior hardware and very fast software on machines where multiplications are costly (in terms of CPU cycles).

Fig. 8 illustrates a flow chart 800 of the method used to design the fast DCT method using scaled terms according to the present invention. In Fig. 8, the one dimensional transform coefficients equations for the DCTs are obtained 810. To obtain the coefficients, the coefficients are split into even and odd terms by obtaining sums and differences with the eight input samples, which are then used to calculate the one dimensional transform coefficients.

After the one dimensional coefficients are obtained, the constants in the 1-D transform matrix are scaled by dividing by one of the constants 820. This reduces the number of multiplications, for an 8x8 block, by four. After scaling, the ratios of constants are very nearly sums of powers-of-2 after only a few summations. Thus, multiplications can be replaced by a series of linear shifts and additions 830.

Fig. 9 illustrates a block diagram 900 of a printer 920 according to the present invention. In Fig. 9, the printer 920 receives data 912 from a host processor 910. The data 912 is provided into memory 930 where the data 912 may be arranged into blocks. The blocks are then processed by a processor 940, such as a raster image processor. The processor 940 provides a compressed print stream representing the data to a printhead driving circuit 950. The printhead driving circuit 950 then controls the printhead 960 to generate a printout 970 of the data.

The algorithm designed with reference to Fig. 8 may be tangibly embodied in a computer-readable medium or carrier 990, e.g. one or more of the fixed and/or

removable data storage devices illustrated in Fig. 9, or other data storage or data communications devices. The computer program may be loaded into the memory 942 to configure the processor 940 of Fig. 9, for execution. The computer program comprise instructions which, when read and executed by the processor 940 of Fig. 9, causes the processor 940 to perform the steps necessary to execute the steps or elements of the present invention.

Fig. 10 illustrates a data analyzing system 1000 according to the present invention. In Fig. 10, a discrete cosine transform 1010 receives a block of data 1012 to be analyzed. The discrete cosine transform 1010 uses discrete cosine transform equations 1020 to generate transformed data 1024. Prior to execution, the discrete cosine transform equations 1020 are split into at least one sub-transform having at least two transform constants. The at least two transform constants for each collection are independently scaled with a scaling term to maintain a substantially uniform ratio between the at least two transform constants within the at least one collection, wherein the scaling term is chosen according to a predetermined cost function. The discrete cosine transformed data 1024 may then be quantized by an optional quantizer 1030.

Fig. 11 illustrates another data analyzing system 1100 according to the present invention. In Fig. 11, a discrete cosine transform 1110 receives a block of data 1112 to be analyzed. The discrete cosine transform 1110 uses discrete cosine transform equations 1120 to generate transformed data 1124. Prior to execution, the discrete cosine transform equations 1120 are split into at least one sub-transform having at least two transform constants. The at least two transform

constants for each collection are independently scaled with a scaling term to maintain a substantially uniform ratio between the at least two transform constants within the at least one collection, wherein the scaling term may be chosen according to a predetermined cost function. The discrete cosine transformed data 1124 may

5 then be compared to scaled comparison values in an optional comparator 1130.

The foregoing description of the exemplary embodiment of the invention has been presented for the purposes of illustration and description. It is not intended to be exhaustive or to limit the invention to the precise form disclosed. Many modifications and variations are possible in light of the above teaching. It is

10 intended that the scope of the invention be limited not with this detailed description, but rather by the claims appended hereto.

11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
1001
1002
1003
1004
1005
1006
1007
1008
1009
1010
1011
1012
1013
1014
1015
1016
1017
1018
1019
1020
1021
1022
1023
1024
1025
1026
1027
1028
1029
1030
1031
1032
1033
1034
1035
1036
1037
1038
1039
1040
1041
1042
1043
1044
1045
1046
1047
1048
1049
1050
1051
1052
1053
1054
1055
1056
1057
1058
1059
1060
1061
1062
1063
1064
1065
1066
1067
1068
1069
1070
1071
1072
1073
1074
1075
1076
1077
1078
1079
1080
1081
1082
1083
1084
1085
1086
1087
1088
1089
1090
1091
1092
1093
1094
1095
1096
1097
1098
1099
1100
1101
1102
1103
1104
1105
1106
1107
1108
1109
1110
1111
1112
1113
1114
1115
1116
1117
1118
1119
1120
1121
1122
1123
1124
1125
1126
1127
1128
1129
1130
1131
1132
1133
1134
1135
1136
1137
1138
1139
1140
1141
1142
1143
1144
1145
1146
1147
1148
1149
1150
1151
1152
1153
1154
1155
1156
1157
1158
1159
1160
1161
1162
1163
1164
1165
1166
1167
1168
1169
1170
1171
1172
1173
1174
1175
1176
1177
1178
1179
1180
1181
1182
1183
1184
1185
1186
1187
1188
1189
1190
1191
1192
1193
1194
1195
1196
1197
1198
1199
1200
1201
1202
1203
1204
1205
1206
1207
1208
1209
1210
1211
1212
1213
1214
1215
1216
1217
1218
1219
1220
1221
1222
1223
1224
1225
1226
1227
1228
1229
1230
1231
1232
1233
1234
1235
1236
1237
1238
1239
1240
1241
1242
1243
1244
1245
1246
1247
1248
1249
1250
1251
1252
1253
1254
1255
1256
1257
1258
1259
1260
1261
1262
1263
1264
1265
1266
1267
1268
1269
1270
1271
1272
1273
1274
1275
1276
1277
1278
1279
1280
1281
1282
1283
1284
1285
1286
1287
1288
1289
1290
1291
1292
1293
1294
1295
1296
1297
1298
1299
1300
1301
1302
1303
1304
1305
1306
1307
1308
1309
1310
1311
1312
1313
1314
1315
1316
1317
1318
1319
1320
1321
1322
1323
1324
1325
1326
1327
1328
1329
1330
1331
1332
1333
1334
1335
1336
1337
1338
1339
1340
1341
1342
1343
1344
1345
1346
1347
1348
1349
1350
1351
1352
1353
1354
1355
1356
1357
1358
1359
1360
1361
1362
1363
1364
1365
1366
1367
1368
1369
1370
1371
1372
1373
1374
1375
1376
1377
1378
1379
1380
1381
1382
1383
1384
1385
1386
1387
1388
1389
1390
1391
1392
1393
1394
1395
1396
1397
1398
1399
1400
1401
1402
1403
1404
1405
1406
1407
1408
1409
1410
1411
1412
1413
1414
1415
1416
1417
1418
1419
1420
1421
1422
1423
1424
1425
1426
1427
1428
1429
1430
1431
1432
1433
1434
1435
1436
1437
1438
1439
1440
1441
1442
1443
1444
1445
1446
1447
1448
1449
1450
1451
1452
1453
1454
1455
1456
1457
1458
1459
1460
1461
1462
1463
1464
1465
1466
1467
1468
1469
1470
1471
1472
1473
1474
1475
1476
1477
1478
1479
1480
1481
1482
1483
1484
1485
1486
1487
1488
1489
1490
1491
1492
1493
1494
1495
1496
1497
1498
1499
1500
1501
1502
1503
1504
1505
1506
1507
1508
1509
1510
1511
1512
1513
1514
1515
1516
1517
1518
1519
1520
1521
1522
1523
1524
1525
1526
1527
1528
1529
1530
1531
1532
1533
1534
1535
1536
1537
1538
1539
1540
1541
1542
1543
1544
1545
1546
1547
1548
1549
1550
1551
1552
1553
1554
1555
1556
1557
1558
1559
1560
1561
1562
1563
1564
1565
1566
1567
1568
1569
1570
1571
1572
1573
1574
1575
1576
1577
1578
1579
1580
1581
1582
1583
1584
1585
1586
1587
1588
1589
1590
1591
1592
1593
1594
1595
1596
1597
1598
1599
1600
1601
1602
1603
1604
1605
1606
1607
1608
1609
1610
1611
1612
1613
1614
1615
1616
1617
1618
1619
1620
1621
1622
1623
1624
1625
1626
1627
1628
1629
1630
1631
1632
1633
1634
1635
1636
1637
1638
1639
1640
1641
1642
1643
1644
1645
1646
1647
1648
1649
1650
1651
1652
1653
1654
1655
1656
1657
1658
1659
1660
1661
1662
1663
1664
1665
1666
1667
1668
1669
1670
1671
1672
1673
1674
1675
1676
1677
1678
1679
1680
1681
1682
1683
1684
1685
1686
1687
1688
1689
1690
1691
1692
1693
1694
1695
1696
1697
1698
1699
1700
1701
1702
1703
1704
1705
1706
1707
1708
1709
1710
1711
1712
1713
1714
1715
1716
1717
1718
1719
1720
1721
1722
1723
1724
1725
1726
1727
1728
1729
1730
1731
1732
1733
1734
1735
1736
1737
1738
1739
1740
1741
1742
1743
1744
1745
1746
1747
1748
1749
1750
1751
1752
1753
1754
1755
1756
1757
1758
1759
1760
1761
1762
1763
1764
1765
1766
1767
1768
1769
1770
1771
1772
1773
1774
1775
1776
1777
1778
1779
1780
1781
1782
1783
1784
1785
1786
1787
1788
1789
1790
1791
1792
1793
1794
1795
1796
1797
1798
1799
1800
1801
1802
1803
1804
1805
1806
1807
1808
1809
1810
1811
1812
1813
1814
1815
1816
1817
1818
1819
1820
1821
1822
1823
1824
1825
1826
1827
1828
1829
1830
1831
1832
1833
1834
1835
1836
1837
1838
1839
1840
1841
1842
1843
1844
1845
1846
1847
1848
1849
1850
1851
1852
1853
1854
1855
1856
1857
1858
1859
1860
1861
1862
1863
1864
1865
1866
1867
1868
1869
1870
1871
1872
1873
1874
1875
1876
1877
1878
1879
1880
1881
1882
1883
1884
1885
1886
1887
1888
1889
1890
1891
1892
1893
1894
1895
1896
1897
1898
1899
1900
1901
1902
1903
1904
1905
1906
1907
1908
1909
1910
1911
1912
1913
1914
1915
1916
1917
1918
1919
1920
1921
1922
1923
1924
1925
1926
1927
1928
1929
1930
1931
1932
1933
1934
1935
1936
1937
1938
1939
1940
1941
1942
1943
1944
1945
1946
1947
1948
1949
1950
1951
1952
1953
1954
1955
1956
1957
1958
1959
1960
1961
1962
1963
1964
1965
1966
1967
1968
1969
1970
1971
1972
1973
1974
1975
1976
1977
1978
1979
1980
1981
1982
1983
1984
1985
1986
1987
1988
1989
1990
1991
1992
1993
1994
1995
1996
1997
1998
1999
2000
2001
2002
2003
2004
2005
2006
2007
2008
2009
2010
2011
2012
2013
2014
2015
2016
2017
2018
2019
2020
2021
2022
2023
2024
2025
2026
2027
2028
2029
2030
2031
2032
2033
2034
2035
2036
2037
2038
2039
2040
2041
2042
2043
2044
2045
2046
2047
2048
2049
2050
2051
2052
2053
2054
2055
2056
2057
2058
2059
2060
2061
2062
2063
2064
2065
2066
2067
2068
2069
2070
2071
2072
2073
2074
2075
2076
2077
2078
2079
2080
2081
2082
2083
2084
2085
2086
2087
2088
2089
2090
2091
2092
2093
2094
2095
2096
2097
2098
2099
2100
2101
2102
2103
2104
2105
2106
2107
2108
2109
2110
2111
2112
2113
2114
2115
2116
2117
2118
2119
2120
2121
2122
2123
2124
2125
2126
2127
2128
2129
2130
2131
2132
2133
2134
2135
2136
2137
2138
2139
2140
2141
2142
2143
2144
2145
2146
2147
2148
2149
2150
2151
2152
2153
2154
2155
2156
2157
2158
2159
2160
2161
2162
2163
2164
2165
2166
2167
2168
2169
2170
2171
2172
2173
2174
2175
2176
2177
2178
2179
2180
2181
2182
2183
2184
2185
2186
2187
2188
2189
2190
2191
2192
2193
2194
2195
2196
2197
2198
2199
2200
2201
2202
2203
2204
2205
2206
2207